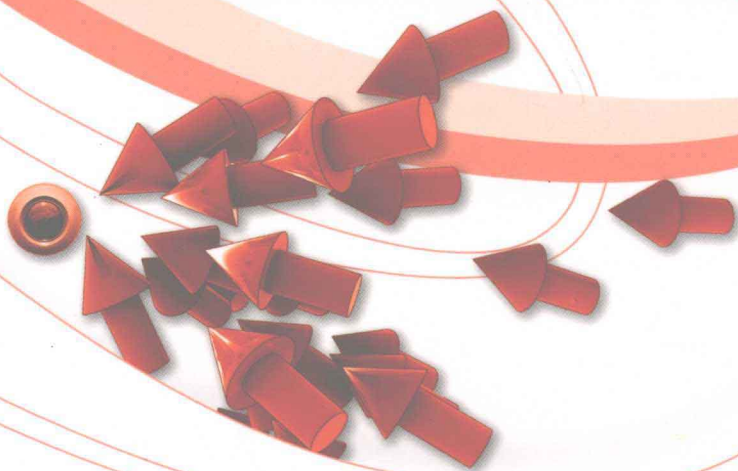




普通高等教育“十二五”规划教材



高等学校计算机规划教材

# C#程序设计基础

## ——教程、实验、习题

■ 赵 敏 主编  
■ 方 芳 万在红 莫 燕 张 帆 廖 远 副主编



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

普通高等教育“十二五”规划教材

高等学校计算机规划教材

# C#程序设计基础

## ——教程、实验、习题

赵 敏 主编

方 芳 万在红 莫 燕 张 帆 廖 远 副主编

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

C#是 Microsoft .NET 的核心编程语言,具有简洁、灵活、安全、面向对象、兼容性等特点,同时具有 Delphi 的高效性和 Visual C++ 的强大功能,是专门针对 .NET 设计的一种语言。本书共 11 章,第 1~5 章介绍 C# 语言程序设计基础,将 C# 语言的语法、知识点循序渐进进行介绍,同时提供了丰富的例题,使初学者能够很快入门。第 6、9~11 章介绍了在 .NET 平台上如何使用 C# 语言来开发各种应用程序,如 Windows 应用程序、数据库编程、Web 应用程序等。第 7 章介绍 GDI+ 的常用类、结构、枚举绘制图形和图像的基本技术。第 8 章介绍文件的操作。各章均配有若干实验,每个实验都包括实验目的、实验要求、完整代码等,对初学者全面掌握本课程大有帮助。本书配有 PPT、例程源代码、习题库(考试系统)、习题解答等相关教学资源。

本书适合作为 Visual C#.NET 程序设计教程,也可作为 Visual C#.NET 开发人员的参考资料。本书不仅适合大中专院校相关专业的学生使用,也可以作为软件培训班的教材或辅助教材使用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容  
版权所有·侵权必究

## 图书在版编目(CIP)数据

C#程序设计基础:教程、实验、习题/赵敏主编. —北京:电子工业出版社,2011.8  
高等学校计算机规划教材  
ISBN 978-7-121-13747-1

I. ①C… II. ①赵… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 103360 号

策划编辑:史鹏举

责任编辑:史鹏举 特约编辑:高采平

印 刷:涿州市京南印刷厂

装 订:涿州市桃园装订有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:16.25 字数:469 千字

印 次:2011 年 8 月第 1 次印刷

定 价:33.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlts@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

# 前 言

C#是一门崭新的面向对象编程语言,具有简单易用、灵活、功能强大、安全性好等特点。通过本书的学习,可使学生掌握使用 C#语言设计应用程序的基本技能,了解面向对象的程序设计方法,能够编写实用、规范、可读性好的应用程序,并为后续课程提供有力的支持,为学生的专业课程设计、毕业设计等实训环节奠定良好的基础。

本书共 11 章,每章包含教程、习题、实验三部分。第一部分在介绍理论知识的基础上,提供了作者多年教学中积累的大量实例,真正做到理论与实践的有机结合;第二部分收集了典型习题;第三部分是上机实验指导,介绍了 C#程序设计语言实验环境,并精心安排设计了多个实验,每个实验都包括详细的实验目的、实验基本要求、实验步骤等,这些实验对初学者全面掌握面向对象的程序设计大有帮助。

第 1~4 章主要介绍 C#的基本语法。其中,第 1、2 章介绍 C#概述、Visual C#.NET 集成开发环境,以及简单的小程序,从而初步了解、认识 C#;第 3 章为 C#程序设计基础,介绍 C#的基本语法,包括 C#数据类型、变量、常量及运算符的使用;第 4 章介绍 C#程序设计的控制语句,详细介绍选择语句、循环语句、跳转语句及数组的使用。第 5 章介绍面向对象程序设计的基础知识,包括类、构造函数、析构函数、方法、属性、方法的参数、方法的重载、类的继承与多态、接口、委托、事件、异常处理、集合与索引器。第 6 章介绍 Windows 应用程序开发,包括 Label、TextBox、Button 等控件的继承及事件处理的内容。第 7 章介绍使用 GDI+中的常用类、结构和枚举绘制图形与图像的基本技术。第 8 章介绍流、目录、文件的常见使用方法。第 9 章介绍数据库应用开发,包括建立与数据库的连接,对数据库中的数据进行查询、插入、删除和更新操作等内容,实验中设计了一个成语词典查询系统综合型实验。第 10 章介绍多线程技术的使用。第 11 章介绍 ASP.NET Web 应用程序开发,介绍了使用 C#语言在 Web 编程中的应用。

本教材吸收了国内外同类教材的长处,版式简洁,方便阅读又节省篇幅,同时对程序代码做了大量的注释,目的在于进一步方便初学者理解,并使其快速进入实际开发角色。书中所有习题及上机实验,均是多年来理论与实践教学中归纳总结出来的,实例丰富,语言通俗易懂,内容循序渐进、深入浅出,形式多样,是针对每章教程的重点、难点而编写的。通过本书的学习,可以加深对理论基础的理解,并在短时间内轻松掌握该门技术。

本书适合作为 Visual C#.NET 程序设计教程,也可作为 Visual C#.NET 开发人员的参考资料。本书不仅适合大中专院校相关专业的学生使用,也可以作为软件培训班的教材或辅助教材使用。

**本书配有 PPT、例程源代码、习题库(考试系统)、习题解答等教学资源**,需要者可登录电子工业出版社华信教育资源网 [www.hxedu.com.cn](http://www.hxedu.com.cn),免费注册、下载。

本书由赵敏主编,方芳、万在红、莫燕、张帆、廖远副主编。闵高鹏、易慧刚、潘竖等对本书的校对付出大量的工作,在此表示感谢。在编写过程中,难免出现差错和不妥之处,欢迎读者批评指正,谢谢!

编 者

# 目 录

## 第1章 C#概述及 Visual Studio.NET 集成

### 开发环境 (IDE) ..... (1)

#### 1.1 .NET 框架概述 ..... (1)

##### 1.1.1 .NET 的定义 ..... (1)

##### 1.1.2 .NET 开发平台 ..... (1)

##### 1.1.3 .NET Framework 结构 ..... (1)

##### 1.1.4 Microsoft Visual Studio.NET (VS.NET) 集成开发环境 (IDE) ..... (2)

#### 1.2 C#语言概述 ..... (2)

##### 1.2.1 C#语言简介 ..... (2)

##### 1.2.2 C#语言特点 ..... (2)

##### 1.2.3 C#语言开发应用范围 ..... (3)

#### 1.3 配置 Visual C#开发环境 ..... (3)

##### 1.3.1 Visual Studio 2008 的安装 ... (3)

##### 1.3.2 C#的启动 ..... (5)

##### 1.3.3 起始页 ..... (6)

##### 1.3.4 Visual C#的退出 ..... (7)

##### 1.3.5 Visual C#菜单栏、工具栏、 工具箱 ..... (7)

##### 1.3.6 解决方案资源管理器 ..... (7)

##### 1.3.7 类视图 ..... (8)

##### 1.3.8 属性窗口 ..... (8)

##### 1.3.9 窗体设计器、代码编辑窗口 ... (8)

#### 习题 1 ..... (9)

#### 上机实验一 ..... (9)

## 第2章 编制一个简单的 C#应用程序 (13)

### 2.1 第一个控制台应用程序设计

#### 实例 ..... (13)

### 2.2 第一个 Windows 应用程序

#### 实例 ..... (15)

### 2.3 C#项目和解决方案 ..... (17)

### 2.4 应用程序起始点 ..... (17)

### 2.5 配置程序集信息 ..... (18)

#### 习题 2 ..... (18)

#### 上机实验二 ..... (19)

##### 实验 VS2008 开发环境下开发简单的 .NET 应用程序 ..... (19)

## 第3章 C#程序设计基础 ..... (23)

### 3.1 C#的关键字和标识符 ..... (23)

### 3.2 C#的数据类型 ..... (25)

#### 3.2.1 值类型 ..... (25)

#### 3.2.2 引用类型 ..... (26)

#### 3.2.3 值类型与引用类型关系 ..... (28)

### 3.3 C#中的变量和常量 ..... (29)

#### 3.3.1 变量 ..... (29)

#### 3.3.2 常量 ..... (30)

### 3.4 运算符和表达式 ..... (31)

#### 3.4.1 运算符 ..... (31)

#### 3.4.2 表达式 ..... (34)

### 3.5 运算符的优先级与结合性 ..... (34)

### 3.6 格式控制符 ..... (36)

### 3.7 控制台输入与输出 ..... (38)

#### 习题 3 ..... (38)

#### 上机实验三 ..... (38)

##### 实验 1 两数相加 ..... (38)

##### 实验 2 求矩形的周长和面积 ..... (40)

## 第4章 C#控制语句及数组的使用 ..... (42)

### 4.1 选择结构语句 ..... (42)

#### 4.1.1 if 语句 ..... (42)

#### 4.1.2 switch 语句 ..... (45)

### 4.2 循环控制语句 ..... (46)

#### 4.2.1 while 语句 ..... (46)

#### 4.2.2 do while 语句 ..... (46)

#### 4.2.3 for 语句 ..... (47)

### 4.3 跳转语句 ..... (47)

### 4.4 数组 ..... (48)

#### 4.4.1 数组的定义 ..... (48)

#### 4.4.2 数组的初始化 ..... (50)

#### 4.4.3 访问数组中的元素 ..... (51)

|                              |       |                            |       |
|------------------------------|-------|----------------------------|-------|
| 4.4.4 数组与 System.Array ..... | (53)  | 第 6 章 Windows 应用程序开发 ..... | (107) |
| 4.4.5 使用 foreach 语句遍历数组      |       | 6.1 窗体 .....               | (107) |
| 元素 .....                     | (54)  | 6.2 Windows 常用控件的使用 .....  | (110) |
| 习题 4 .....                   | (54)  | 6.2.1 标签控件和超链接标签           |       |
| 上机实验四 .....                  | (55)  | 控件 .....                   | (112) |
| 实验 熟练掌握循环语句 .....            | (55)  | 6.2.2 按钮控件 .....           | (113) |
| 第 5 章 面向对象编程基础 .....         | (56)  | 6.2.3 文本框控件、富文本框           |       |
| 5.1 面向对象概念 .....             | (56)  | 控件 .....                   | (114) |
| 5.2 类 .....                  | (56)  | 6.2.4 单选按钮控件 .....         | (117) |
| 5.2.1 类的声明 .....             | (57)  | 6.2.5 复选框控件 .....          | (118) |
| 5.2.2 类的成员 .....             | (57)  | 6.2.6 列表框控件 .....          | (118) |
| 5.2.3 构造函数 .....             | (58)  | 6.2.7 组合框控件 .....          | (118) |
| 5.2.4 析构函数 .....             | (61)  | 6.2.8 分组框控件 .....          | (119) |
| 5.3 方法 .....                 | (61)  | 6.2.9 面板控件 .....           | (122) |
| 5.3.1 方法的声明 .....            | (62)  | 6.2.10 图片框控件 .....         | (122) |
| 5.3.2 方法的参数 .....            | (63)  | 6.2.11 定时器控件 .....         | (122) |
| 5.3.3 方法的重载 .....            | (66)  | 6.2.12 滚动条控件 .....         | (123) |
| 5.3.4 静态方法和实例方法 .....        | (67)  | 6.2.13 月历控件 .....          | (123) |
| 5.4 属性 .....                 | (68)  | 6.2.14 工具栏控件 .....         | (124) |
| 5.5 类的继承与多态 .....            | (71)  | 6.2.15 状态栏控件 .....         | (125) |
| 5.5.1 类的继承 .....             | (71)  | 6.3 菜单 .....               | (125) |
| 5.5.2 类的多态 .....             | (74)  | 6.3.1 菜单控件 .....           | (125) |
| 5.6 接口 .....                 | (78)  | 6.3.2 上下文菜单 .....          | (127) |
| 5.7 委托与事件 .....              | (81)  | 6.4 对话框 .....              | (127) |
| 5.8 异常处理 .....               | (86)  | 6.5 通用对话框 .....            | (128) |
| 5.9 集合与索引器 .....             | (90)  | 6.5.1 消息框 .....            | (128) |
| 5.9.1 集合 .....               | (90)  | 6.5.2 打开文件对话框 .....        | (129) |
| 5.9.2 索引器 .....              | (93)  | 6.5.3 保存文件对话框 .....        | (131) |
| 习题 5 .....                   | (94)  | 6.5.4 颜色对话框 .....          | (131) |
| 上机实验五 .....                  | (95)  | 6.5.5 字体对话框 .....          | (132) |
| 实验 1 抽象类和抽象方法的               |       | 6.6 多文档界面 .....            | (133) |
| 使用 .....                     | (95)  | 6.6.1 创建 MDI 主窗体           |       |
| 实验 2 接口演示 .....              | (96)  | (父窗体) .....                | (133) |
| 实验 3 接口实现 .....              | (97)  | 6.6.2 创建 MDI 子窗体 .....     | (133) |
| 实验 4 虚方法与非虚方法的               |       | 习题 6 .....                 | (134) |
| 区别 .....                     | (99)  | 上机实验六 .....                | (135) |
| 实验 5 密封类和密封方法的               |       | 实验 1 设计程序, 实现 A! +B! +C! 的 |       |
| 使用 .....                     | (100) | 运算并输出运算结果 .....            | (135) |
| 实验 6 委托的声明和使用 .....          | (102) | 实验 2 设计 Windows 应用程序, 根据   |       |
| 实验 7 异常处理 .....              | (104) | 单选按钮和复选框的选择, 分别显           |       |
|                              |       | 示时间和日期 .....               | (137) |

|                                                                             |       |                                                 |       |
|-----------------------------------------------------------------------------|-------|-------------------------------------------------|-------|
| 实验 3 创建一个 ListBox .....                                                     | (139) | 9.1.2 结构化查询语言(SQL) .....                        | (185) |
| 实验 4 创建一个菜单驱动, 实现<br>最简单文字编辑功能<br>的记事本程序 .....                              | (140) | 9.2 ADO.NET 概述 .....                            | (189) |
| 第 7 章 GDI+编程 .....                                                          | (145) | 9.3 创建连接 .....                                  | (190) |
| 7.1 创建 Graphics 对象 .....                                                    | (145) | 9.4 使用 Command 对象与 DataReader<br>对象 .....       | (192) |
| 7.2 画笔 .....                                                                | (147) | 9.4.1 Command 对象 .....                          | (192) |
| 7.3 画刷 .....                                                                | (148) | 9.4.2 DataReader 对象 .....                       | (194) |
| 7.4 颜色 .....                                                                | (152) | 9.5 使用 DataSet 对象与 DataAdapter<br>对象 .....      | (197) |
| 7.5 绘制线条或空心形状 .....                                                         | (152) | 9.5.1 DataSet 对象 .....                          | (197) |
| 7.6 绘制实心形状 .....                                                            | (153) | 9.5.2 DataAdapter 对象 .....                      | (199) |
| 7.7 用 GDI+绘制文本 .....                                                        | (153) | 9.6 数据绑定 .....                                  | (205) |
| 7.7.1 字体 .....                                                              | (153) | 9.6.1 数据绑定概述 .....                              | (205) |
| 7.7.2 文本处理 .....                                                            | (154) | 9.6.2 简单数据绑定 .....                              | (205) |
| 7.8 用 GDI+显示图像 .....                                                        | (155) | 9.6.3 复杂数据绑定 .....                              | (206) |
| 习题 7 .....                                                                  | (157) | 9.7 DataGridView 控件 .....                       | (209) |
| 上机实验七 .....                                                                 | (157) | 习题 9 .....                                      | (211) |
| 实验 设计一个简易 Windows 绘图板<br>(利用 Graphics 对象绘制线条和<br>形状、呈现文本、显示或操作<br>图像) ..... | (157) | 上机实验九 .....                                     | (212) |
| 第 8 章 文件操作 .....                                                            | (166) | 实验 设计一个成语词典查询<br>系统 .....                       | (212) |
| 8.1 流 .....                                                                 | (166) | 第 10 章 C#多线程技术 .....                            | (220) |
| 8.1.1 文本文件的读取和写入 .....                                                      | (167) | 10.1 线程概述 .....                                 | (220) |
| 8.1.2 二进制流的读/写 .....                                                        | (168) | 10.2 .NET 对多线程的支持 .....                         | (220) |
| 8.1.3 文件流的读/写 .....                                                         | (170) | 10.3 一个多线程程序 .....                              | (221) |
| 8.2 文件类 .....                                                               | (171) | 10.4 线程的优先级 .....                               | (223) |
| 8.3 目录类 .....                                                               | (175) | 10.5 线程同步 .....                                 | (225) |
| 8.4 Path 类 .....                                                            | (178) | 习题 10 .....                                     | (232) |
| 8.5 创建文件 .....                                                              | (179) | 上机实验十 .....                                     | (233) |
| 8.6 读/写文件 .....                                                             | (179) | 实验 1 创建两个线程, 打印<br>从 1 到 10 的数字<br>(不带参数) ..... | (233) |
| 8.6.1 写入文件 .....                                                            | (179) | 实验 2 创建两个线程, 打印<br>从 1 到 10 的数字<br>(带参数) .....  | (234) |
| 8.6.2 读取文件 .....                                                            | (180) | 实验 3 使用 lock 同步线程 .....                         | (235) |
| 习题 8 .....                                                                  | (180) | 第 11 章 ASP.NET Web 应用程序<br>开发 .....             | (237) |
| 上机实验八 .....                                                                 | (181) | 11.1 ASP.NET 简介 .....                           | (237) |
| 实验 简易文件管理器制作 .....                                                          | (181) | 11.1.1 ASP.NET 的开发环境<br>配置 .....                | (237) |
| 第 9 章 数据库应用开发 .....                                                         | (184) |                                                 |       |
| 9.1 数据库概述 .....                                                             | (184) |                                                 |       |
| 9.1.1 关系数据库模型 .....                                                         | (184) |                                                 |       |

|                      |       |
|----------------------|-------|
| 11.1.2 简单网站的创建       | (237) |
| 11.2 Web Form        | (239) |
| 11.3 HTML 控件         | (240) |
| 11.3.1 HTML(超文本标记语言) |       |
| 简介                   | (240) |
| 11.3.2 HTML 控件的使用    | (241) |
| 11.4 服务器控件           | (242) |
| 11.4.1 标准控件          | (242) |
| 11.4.2 验证控件          | (244) |

|             |       |
|-------------|-------|
| 11.4.3 数据操作 | (246) |
| 习题 11       | (247) |
| 上机实验十一      | (247) |

|                             |       |
|-----------------------------|-------|
| 实验 创建 ASP.NET 页面 Grid.aspx, |       |
| 使用 GridView 控件显示范例          |       |
| 数据库 Northwind 中 Products    |       |
| 表的数据信息                      | (248) |

|      |       |
|------|-------|
| 参考文献 | (251) |
|------|-------|

# 第1章 C#概述及 Visual Studio.NET 集成开发环境(IDE)

## 1.1 .NET 框架概述

### 1.1.1 .NET 的定义

.NET 技术是微软公司推出的一个全新概念,“它代表了一个集合、一个环境和一个可以作为平台支持下一代 Internet 的可编程结构”。

通过.NET 这个平台,可以使用多种语言开发同一个项目,实现这些语言的跨平台应用。而微软默认的语言是 Visual C#.NET。

### 1.1.2 .NET 开发平台

.NET 开发平台包括.NET 框架和.NET 开发工具等组成部分,.NET 框架(Framework)是整个开发平台的基础,包括公共语言运行库和框架类库,.NET 开发工具包括 Visual Studio.NET 集成开发环境和.NET 编程语言。

### 1.1.3 .NET Framework 结构

.NET Framework(框架)的组成:公共语言运行时环境(CLR);基类库(BCL);基于 ASP.NET 编程框架的网络服务和网络表单;Windows 桌面应用界面编程组件;数据库访问组件。

它的版本有:

- 1.x: 已用于 Visual Studio 2003;
- 2.0: 已用于 Visual Studio 2005;
- 3.5: 已用于 Visual Studio 2008;
- 4.0: 已用于 Visual Studio 2010。

Microsoft .NET Framework 是一个用于 Windows 应用程序、Web 应用程序、控制台应用程序和智能设备应用程序的平台。.NET Framework 提供丰富的类库,程序员可以使用类库来减少需要编写、测试和维护的代码量。

.NET 技术的核心是.NET Framework,它是构建于计算机网络基础上的开发工具。.NET Framework 的基本结构如图1.1所示。

.NET Framework 的两个主要组件,公共语言运行时环境和.NET Framework 类库。

#### (1) 公共语言运行时环境

.NET 框架的底层是公共语言运行时环境,提供了程序代码可以跨平台执行的机制,另外还提供了内存管理、线程管理、远程处理等核心服务,并还强制实施严格的类型安全检查操作,从而提高代码的安全性、可靠性和准确性。它提供了系统资源统一管理和安全机制。其中,以公共语言运行库为目标的代码为“托管代码”,而不以公共语言运行库为目标的代码称为“非托管代码”。

#### (2) .NET Framework 类库

ADO.NET、Windows Forms 和 ASP.NET 等组件库不是 C#或 Visual Basic.NET 语言的一部分,而是进行相应的应用软件开发所必须掌握的。

所有 .NET 开发语言的能力和执行速度基本相同；

只要遵守公共语言规范 (Common Language Specification, CLS) 和支持公共语言运行时 (Common Language Runtime, CLR)，可以采用其他语言；

.NET 语言生成的都是托管代码，这种代码在运行库的控制之下运行；

.NET 编译生成的可执行文件是一种托管代码，称为微软中间语言 (Microsoft Intermediate Language, MSIL)，可以“一次编写，随处运行”，前提是有框架的支持。

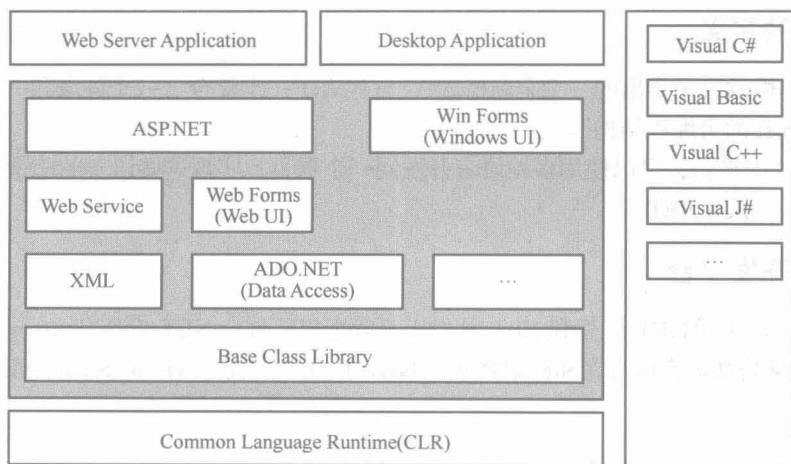


图 1.1 .NET 框架的组成

#### 1.1.4 Microsoft Visual Studio.NET (VS.NET) 集成开发环境 (IDE)

Microsoft Visual Studio.NET (VS.NET) 是一套完整的开发工具，是一个集成开发环境 (IDE)，用于生成 Windows 应用程序、ASP.NET Web 应用程序、XML Web Services 和移动应用程序。其中包含的 Visual Basic.NET、Visual C++.NET、Visual C#.NET 等，全都使用相同的集成开发环境，该环境允许它们共享工具，并有助于创建混合语言解决方案。

### 1.2 C#语言概述

#### 1.2.1 C#语言简介

C#起源于 C 语言家族，因此具有 C++ 的功能。C#采用与 C、C++ 或 Java 一致的花括号 ({} ) 语法，简单易学。此外，C#语法简化了 C++ 的诸多复杂性，同时又提供了 Java 所不具备的很多强大的功能，例如，可为 null 的值类型、枚举、委托、lambda 表达式和直接内存访问。

C#语言是随 .NET 一起设计出来的全新的开发语言，其设计目的就是作为 VS.NET 的核心语言。经过几年的发展，C#已经成为 Windows 平台上软件开发的主流语言之一。

#### 1.2.2 C#语言特点

C#是一种现代的、面向对象的、类型安全的编程语言。C#具有下列特点：

(1) 简单。C#简化了 C/C++ 中许多复杂的特性。例如，C#中禁止直接使用指针，从而避免了直接操作内存的复杂性和风险性；类的成员调用统一采用“.”操作符，避免了 C++ 中的“::”或“->”

操作符的复杂性;布尔值是纯粹的 True 和 False 的值,而不是容易产生歧义的整型值;采用“==”比较操作,从而避免 C 语言中与赋值操作“=”的混淆错误。

(2) 面向对象。C#支持数据封装、继承、多态和接口。所有的变量和方法,包括 Main 方法,都封装在类定义中,类可能直接从一个父类继承(不支持多重继承),但可以实现任意数量的接口,所有 C#类型都继承于唯一的根类型 object。原始数据类型(如 int 和 double)通过装箱和拆箱操作可以与对象类型(如 Integer 和 Double)之间进行相互转换。

(3) 类型安全。C#是强类型语言,即每个变量和对象都必须具有声明类型。C#禁止进行不安全的类型转换。数组类型下标从 0 开始而且进行越界检查。另外,C#中类型溢出将被检查。C#支持泛型类型和方法,进而提供了更为出色的类型安全和性能。

(4) 现代。C#语言包括许多现代语言特点,例如:

- 支持属性,充当私有成员变量的访问器。
- 支持封装的方法签名(称为“委托”),实现了类型安全的事件通知。
- 分部方法定义。分部类型可以包含分部方法。
- 支持内联 XML 文档注释,编入了自己的文档。
- 支持泛型方法和类型,从而提供更出色的类型安全和性能。

(5) 兼容性。虽然 C#语言中不提倡使用指针数据类型,但在 C#代码中仍然允许使用需要传递指针参数的 API 交互操作,从而保证与 C/C++代码的兼容性。

### 1.2.3 C#语言开发应用范围

C#语言主要用来构建在 .NET Framework 上运行的各种安全、可靠的应用程序。可创建下列类型的应用程序和服务:

- 桌面应用。包括控制台应用程序、Windows 窗体应用程序等。
- Web 应用。包括 ASP.NET 应用程序、Web 服务等。
- Windows Presentation Foundation (WPF) 应用程序。
- 面向服务的应用程序 (Windows Communication Foundation, WCF)。
- 工作流程的应用程序 (Windows Workflow Foundation, WWF)。
- 移动应用。
- Office 平台应用程序。
- Windows 服务。

## 1.3 配置 Visual C#开发环境

### 1.3.1 Visual Studio 2008 的安装

将安装盘放入光驱后,会自动运行安装程序,出现如图 1.2 所示的安装界面。

选择“安装 Visual Studio 2008”,出现安装向导界面,如图 1.3 所示。然后单击“下一步”按钮,显示“选择要安装的功能”及“产品安装路径”,如图 1.4 所示。

在“选择要安装的功能”中选择“自定义”,单击“产品安装路径”的“浏览”按钮可更改安装文件路径,路径应为英文路径。选择好所安装的路径后,单击“安装”按钮,显示“选择要安装的功能”的下一个界面,如图 1.5 所示,在该页面中,可以选择要安装具体哪些语言等选项,然后单击“安装”按钮,将自动安装完成。



图 1.2 Visual Studio 2008 的安装界面

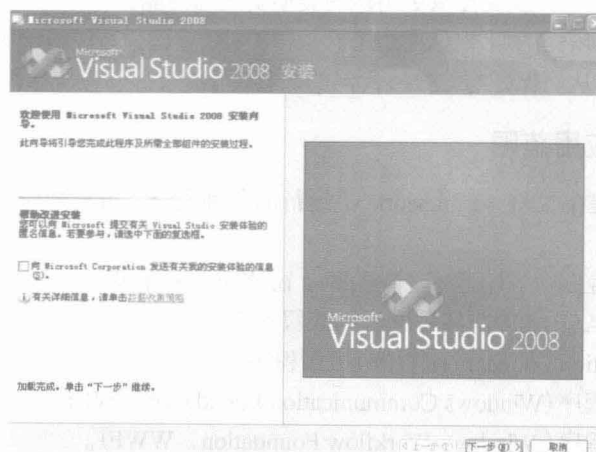


图 1.3 安装向导



图 1.4 “选择要安装的功能”及“产品安装路径”

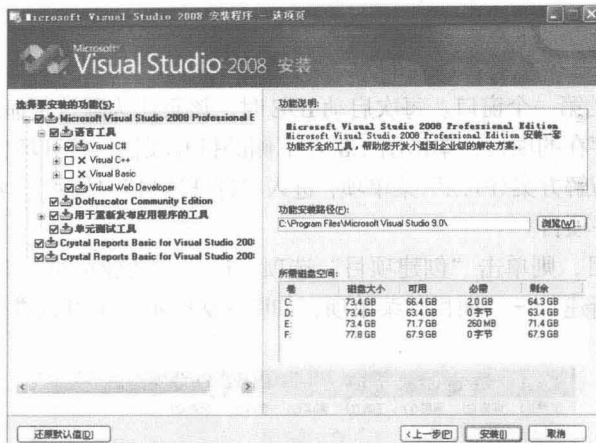


图 1.5 选择要安装的功能

### 1.3.2 C#的启动

安装完 Microsoft Visual Studio 2008 后, 选择“开始”→“程序”→“Microsoft Visual Studio 2008”→“Microsoft Visual Studio 2008”, Microsoft Visual Studio 2008 就启动了, 如图 1.6 所示。



图 1.6 开始菜单下 Microsoft Visual Studio 2008 启动界面

第一次启动 Microsoft Visual Studio 2008 会出现如图 1.7 所示“选择默认环境设置”界面。在该页面中, 选择“Visual C#开发设置”选项, 并单击“启动 Visual Studio”按钮, 进入起始页, 如图 1.8 所示。

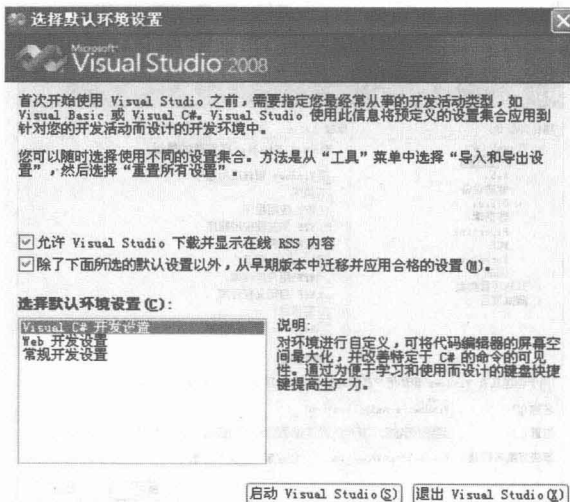


图 1.7 选择默认环境设置界面

### 1.3.3 起始页

起始页是打开 IDE 的第一个窗口。每次启动 IDE 时, 该页作为默认页显示。

若想打开某个已经存在的项目, 单击图 1.8 左下侧的打开项目。打开项目的另一方法: 选择“文件”→“打开”→“项目/解决方案(P)…”菜单项, 进入“打开项目”对话框。选中某项目后, 单击“打开”按钮, 就打开了某一项目。

若要创建一个新项目, 则单击“创建项目”选项, 进入“新建项目”对话框。新建项目另一方法: 选择“文件”→“新建”→“项目”菜单项, 如图 1.9 所示, 也可以进入“新建项目”对话框, 如图 1.10 所示。



图 1.8 起始页界面

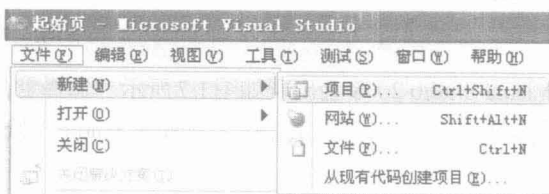


图 1.9 用菜单方法新建项目



图 1.10 “新建项目”对话框

在“新建项目”对话框中, 选中所创建项目的模板, 输入项目名称及存储项目的位置等, 再单击“确定”按钮, 就创建好了一个空的项目。

### 1.3.4 Visual C#的退出

选择“文件”→“退出”菜单项, 就可退出 Visual C#环境, 如图 1.11 所示。



图 1.11 退出 Visual C#环境

### 1.3.5 Visual C#菜单栏、工具栏、工具箱

如图 1.12 所示, 最上面一栏为标题栏。标题栏下面为菜单栏, 再向下是工具栏。标题栏显示项目的标题; 菜单栏中的菜单项可以实现 Visual Studio 的各种功能; 而工具栏则是实现 Visual Studio 常用的功能。

如图 1.13 所示是工具箱界面。按下工具箱右上角中间的图标, 则把工具箱固定, 随时都可见。若再按下图标, 就把工具箱隐藏在整個 IDE 环境的最左侧。只要鼠标停留在隐藏的工具箱图标上就会再次显示工具箱。当鼠标离开工具箱时, 工具箱又会自动隐藏。



图 1.12 标题栏、菜单栏、工具栏

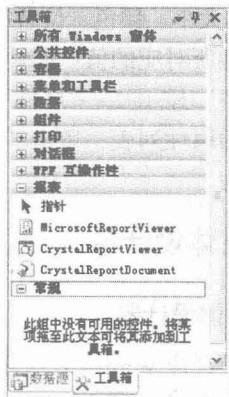


图 1.13 工具箱界面

### 1.3.6 解决方案资源管理器

当创建项目后, 会在“解决方案资源管理器”中显示自动生成的项目, 如图 1.14 所示。解决方案中包含一个或多个项目, 每个项目都对应于软件中的一个模块。在解决方案资源管理器中, 将同类的

文件放在一个目录下，当单击这个目录后，会将对应的目录下的文件全部显示出来。例如，单击“引用”文件夹前面的“+”，就会将引用的程序集全部显示出来，同时“+”变成“-”。

用鼠标右键单击“解决方案资源管理器”中的节点，将弹出一个上下文菜单，通过其中的菜单命令可以对节点对象进行操作。

### 1.3.7 类视图

图 1.14 介绍的解决方案资源管理器是基于文件的项目管理，而 C# 是一种面向对象的编程语言。其编程用得最多的是对象，而对象是归属于类的。为此，Visual Studio 2008 提供了类视图来对项目中的类进行管理，在“解决方案资源管理器”窗口的下端，单击“类视图”页面，就切换到“类视图”窗口。在类视图窗口中可以查看当前项目的类和类型的层次信息，如图 1.15 所示。

其中，

{ }：表示命名空间；

📁：表示基类；

📁：表示普通类或子类。

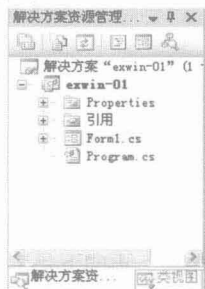


图 1.14 解决方案资源管理器

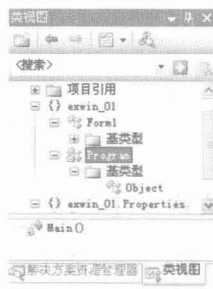


图 1.15 类视图

### 1.3.8 属性窗口

属性窗口 (见图 1.16) 用于操纵一个窗体或控件的属性。属性定义了控件的一些信息，如大小、颜色、位置等。属性窗口左边一栏显示控件的属性，右边一栏显示属性所对应的值。属性窗口有几个工具按钮：“按分类顺序”显示 📁、“字母顺序”显示 🔍、“属性”显示 📋、“事件”显示 ⚡。单击“属性”按钮，显示的是某控件的属性列表；单击“事件”按钮，显示的是某控件或窗体响应特定的用户行为。



图 1.16 属性

### 1.3.9 窗体设计器、代码编辑窗口

#### 1. 窗体设计器

对于设计像 WinForm 这样的用户界面项目，IDE 提供了类似于 Delphi 开发环境下的设计器窗口，即提供了一个图形化用户界面的设计环境，大大提高了开发速度。利用此环境设计一个窗体或服务端对象，设计器会自动修改代码来反映所做的界面修改。选择“视图”→“设计器”菜单项，就可显示设计器窗口，如图 1.17 所示。

#### 2. 代码编辑窗口

选择“视图”→“代码”菜单项；或在设计器中，选中窗体或控件，单击鼠标右键，在弹出的

快捷菜单中选择“查看代码”菜单项；或双击窗体或控件；也可在设计器中按快捷键<F7>，以上几种方法都可以显示代码编辑窗口。

如图 1.18 所示为代码编辑窗口。该窗口顶部的两个下拉列表框可用于选择和浏览相应的代码。左边的列表框包含了代码中的所有类；右边的列表框包含了左边列表框中当前类的所有方法、数据成员、所使用的控件及相应的事件处理方法。选中一项后，光标会定位于对应的代码的第一行。

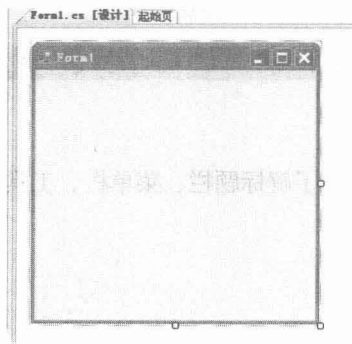


图 1.17 窗体设计器



图 1.18 代码编辑窗口

## 习题 1

- 在 Visual Studio.NET 窗口中，在\_\_\_\_\_窗口中可以查看当前项目的类和类型的层次信息
  - 解决方案资源管理器
  - 类视图
  - 资源视图
  - 属性
- C#语言主要面向\_\_\_\_\_开发环境进行编程。
  - DOS
  - .NET
  - Windows
  - Linux
- 下面关于 C#语言说法中，不正确的是\_\_\_\_\_。
  - C#语言完全支持面向对象编程
  - C#从 C/C++发展而来，但更具有优势，如开发 Web 应用程序和 NT 服务等
  - C#语言简单，编程灵活，但功能有限，不适合开发 Windows 应用程序
  - C#拥有对 .NET 基类库的完全访问权，并易于访问 Windows API

## 上机实验一

实验目的：

熟悉 VS2008 开发环境。

实验要求：

- 通过实验，熟悉 VS2008 开发环境中的标题栏、菜单栏、工具栏。
- 掌握怎样开启工具箱、属性窗口、事件窗口、解决方案资源管理器。
- 通过实验了解如何使用帮助。